CLAIMS

What is claimed is:

1. A method for performing input/output operations on a memory by traversing a queue of commands, comprising:

accessing a queue entry of a queue of commands;

determining if the queue entry includes a valid command; and

if the queue entry includes a valid command, then determining if the valid command is a read command; and

if the valid command is determined to be a read command, determining if a Next Valid Write Address pointer field of the queue entry holds a valid write address pointer.

2. The method of Claim 1, wherein the queue entry aforementioned is a first queue entry, and, further comprising, if the first queue entry includes a Next Valid Write Address pointer, then using the Next Valid Write Address pointer of the queue entry to access a second queue entry.

3. The method of Claim 2, further comprising, after accessing the second queue entry, then commencing to fill a pipeline for the write command corresponding to the second queue entry.

4. The method of Claim 1, further comprising, if the valid command is a read command, then performing the read command.

5. The method of Claim 1, further comprising, before determining if the Next Valid Write Address pointer field of the queue entry holds a valid write address pointer, allowing a concurrent write command to complete.

6. The method of Claim 1, further comprising, before determining if the Next Valid Write Address pointer field of the queue entry holds a valid write address

pointer, terminating any concurrent write commands in progress.

7. The method of Claim 1, further comprising, while determining if the Next Valid Write Address pointer is valid, processing the valid command that is a read command.

8. The method of Claim 7, further comprising, if the Next Valid Write Address pointer is valid, then accessing the queue entry corresponding to the Next Valid Write Address pointer.

9. The method of Claim 8, further comprising, determining if a command at the queue entry corresponding to the Next Valid Write Address pointer is a valid write command.

10. The method of Claim 9, further comprising, if the command at the queue entry is determined be a valid write command, preparing data in an outbound pipeline in anticipation of the valid write command being processed.

11. The method of Claim 10, further comprising, advancing through the queue of entries such that for each just accessed queue entry, a command at a just accessed queue entry is performed.

12. The method of Claim 11, wherein the command at the just accessed queue entry is another read command.

13. The method of Claim 11, wherein the command at the just accessed queue entry is the write command.

14. The method of Claim 11, wherein after the queue of commands has been traversed, any incomplete commands in the queue of commands are processed in another traversal of the queue of commands.

15. The method of Claim 1, wherein the queue of commands is a circular queue buffer.

16. The method of Claim 1, wherein the queue of commands is traversable in a forward direction or a backward direction.

17. The method of Claim 1, wherein the queue of commands is traversable in alternating forward and backward directions.

18. A system for traversing a queue of commands, comprising:

a processor that issues commands, the commands being maintained in a queue of commands, the commands being issued by the processor in successive order from the queue, the processor reissuing commands that are not properly completed;

first and second state machines; and

a memory coupled to the processor through the first and second state machines;

wherein data transfers to and from the memory are initially stored in a queue of commands.

19. The system of Claim 18, wherein commands from the queue of commands are accessed sequentially, each of the commands being one of the group consisting of a read command, a write command, and an instruction that does nothing (no op).

20. The system of Claim 19, wherein all entries of the queue contain a Next Valid Write Address pointer.

21. The system of Claim 20, wherein each command of the queue of commands has an associated field indicating if the queue command is a valid queue command.

22. The system of Claim 21, wherein the associated field is a bit.

23. The system of Claim 22, further comprising a device coupled to the processor and the memory.

24. The system of Claim 23, further comprising a bus for coupling the device to the processor and the memory.

25. The system of Claim 24, wherein the bus is a Peripheral Component

Interconnect (PCI) bus.

26. The system of Claim 25, wherein the PCI bus is a PCI extended (PCI-X) bus.

27. The system of Claim 25, wherein the PCI bus is a PCI Express bus.

28. The system of Claim 23, wherein the device is a redundant array of independent disks.

29. A system for traversing a queue of commands, comprising:

a means for receiving and processing commands, the commands being maintained in a queue of commands;

a means for storing data, the means for storing data being coupled to the means for receiving and processing commands, the means for storing data receiving and providing data from storage in response to commands from the queue of commands,

wherein the queue of commands contains a Next Valid Write Address pointer.

30. The system of Claim 29, further comprising means for transferring data to the means for storing data and means for transferring data from the means for storing data.

31. The system of Claim 30, wherein if an instruction for transferring data to the means for storing data is being processed, an instruction for transferring data from the means for storing data is simultaneously being processed.

32. The system of Claim 31, further comprising a means for processing coupled to the means for storing data via the means for transferring data to the means for storing data and the means for transferring data from the means for storing data.

33. The system of Claim 32, further comprising a means for coupling the means for processing to the means for transferring data to the means for storing data and the means for transferring data from the means for storing data.

34. The system of Claim 33, wherein the means for coupling is a Peripheral Component Interconnect (PCI) bus.

35. The system of Claim 33, wherein the means for coupling is a Peripheral Component Interconnect Extended (PCI-X) bus.

36. The system of Claim 33, wherein the means for coupling is a Peripheral Component Interconnect Express (PCI Express) bus.